F 7

```
BBBBBBBBBBBB        AAAAAAAAA        CCCCCCCCCCCC  KKK         KKK  UUU         UUU  PPPPPPPPPPPP
BBBBBBBBBBBB        AAAAAAAAA        CCCCCCCCCCCC  KKK         KKK  UUU         UUU  PPPPPPPPPPPP
BBBBBBBBBBBB        AAAAAAAAA        CCCCCCCCCCCC  KKK         KKK  UUU         UUU  PPPPPPPPPPPP
BBB         BBB  AAA         AAA  CCC              KKK         KKK  UUU         UUU  PPP         PPP
BBB         BBB  AAA         AAA  CCC              KKK         KKK  UUU         UUU  PPP         PPP
BBB         BBB  AAA         AAA  CCC              KKK         KKK  UUU         UUU  PPP         PPP
BBB         BBB  AAA         AAA  CCC              KKK      KKK     UUU         UUU  PPP         PPP
BBB         BBB  AAA         AAA  CCC              KKK      KKK     UUU         UUU  PPP         PPP
BBB         BBB  AAA         AAA  CCC              KKK      KKK     UUU         UUU  PPP         PPP
BBBBBBBBBBBB     AAA         AAA  CCC              KKKKKKKKK        UUU         UUU  PPPPPPPPPPPP
BBBBBBBBBBBB     AAA         AAA  CCC              KKKKKKKKK        UUU         UUU  PPPPPPPPPPPP
BBBBBBBBBBBB     AAA         AAA  CCC              KKKKKKKKK        UUU         UUU  PPPPPPPPPPPP
BBB         BBB  AAAAAAAAAAAAAAA  CCC              KKK      KKK     UUU         UUU  PPP
BBB         BBB  AAAAAAAAAAAAAAA  CCC              KKK      KKK     UUU         UUU  PPP
BBB         BBB  AAAAAAAAAAAAAAA  CCC              KKK      KKK     UUU         UUU  PPP
BBB         BBB  AAA         AAA  CCC              KKK         KKK  UUU         UUU  PPP
BBB         BBB  AAA         AAA  CCC              KKK         KKK  UUU         UUU  PPP
BBB         BBB  AAA         AAA  CCC              KKK         KKK  UUU         UUU  PPP
BBBBBBBBBBBB     AAA         AAA     CCCCCCCCCCCC  KKK         KKK  UUUUUUUUUUUUUUU  PPP
BBBBBBBBBBBB     AAA         AAA     CCCCCCCCCCCC  KKK         KKK  UUUUUUUUUUUUUUU  PPP
BBBBBBBBBBBB     AAA         AAA     CCCCCCCCCCCC  KKK         KKK  UUUUUUUUUUUUUUU  PPP
```

```
                                         G 14
**FILE**ID**STABACCOP





  SSSSSSSS  TTTTTTTTTT    AAAAAA    BBBBBBBB      AAAAAA      CCCCCCCC    CCCCCCCC    OOOOOO    PPPPPPPP
  SSSSSSSS  TTTTTTTTTT    AAAAAA    BBBBBBBB      AAAAAA      CCCCCCCC    CCCCCCCC    OOOOOO    PPPPPPPP
SS              TT      AA      AA  BB      BB  AA      AA  CC          CC          OO      OO  PP      PP
SS              TT      AA      AA  BB      BB  AA      AA  CC          CC          OO      OO  PP      PP
SS              TT      AA      AA  BB      BB  AA      AA  CC          CC          OO      OO  PP      PP
SS              TT      AA      AA  BB      BB  AA      AA  CC          CC          OO      OO  PP      PP
  SSSSSS        TT      AA      AA  BBBBBBBB    AA      AA  CC          CC          OO      OO  PPPPPPPP
  SSSSSS        TT      AA      AA  BBBBBBBB    AA      AA  CC          CC          OO      OO  PPPPPPPP
        SS      TT      AAAAAAAAAA  BB      BB  AAAAAAAAAA  CC          CC          OO      OO  PP
        SS      TT      AAAAAAAAAA  BB      BB  AAAAAAAAAA  CC          CC          OO      OO  PP
        SS      TT      AA      AA  BB      BB  AA      AA  CC          CC          OO      OO  PP          ....
        SS      TT      AA      AA  BB      BB  AA      AA  CC          CC          OO      OO  PP          ....
SSSSSSSS        TT      AA      AA  BBBBBBBB    AA      AA    CCCCCCCC    CCCCCCCC    OOOOOO    PP          ....
SSSSSSSS        TT      AA      AA  BBBBBBBB    AA      AA    CCCCCCCC    CCCCCCCC    OOOOOO    PP          ....

LL              IIIIII      SSSSSSSS
LL              IIIIII      SSSSSSSS
LL                II      SS
LL                II      SS
LL                II      SS
LL                II      SS
LL                II        SSSSSS
LL                II        SSSSSS
LL                II              SS
LL                II              SS
LL                II              SS
LL                II              SS
LLLLLLLLLL    IIIIII    SSSSSSSS
LLLLLLLLLL    IIIIII    SSSSSSSS
```

```
0 MODULE STABACCOP(%TITLE 'Copy image file for Standalone BACKUP kit'
0                 MAIN = STABACCOP,
0                 IDENT = 'V04-000'
0                 ) =
1 BEGIN
1
1 !
1 !****************************************************************************
1 !*                                                                          *
1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                  *
1 !*  ALL RIGHTS RESERVED.                                                    *
1 !*                                                                          *
1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
1 !*  TRANSFERRED.                                                            *
1 !*                                                                          *
1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
1 !*  CORPORATION.                                                            *
1 !*                                                                          *
1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
1 !*                                                                          *
1 !*                                                                          *
1 !****************************************************************************
1 !
1
1 !++
1 ! FACILITY:
1 !       General utility programs.
1 !
1 ! ABSTRACT:
1 !       This program copies an image file, deleting the appended patch text.
1 !       It is used and supported only for generation of Standalone BACKUP kits.
1 !
1 ! ENVIRONMENT:
1 !       VAX/VMS user mode.
1 !--
1 !
1 ! AUTHOR: M. Jack, CREATION DATE: 16-Sep-1982
1 !
1 ! MODIFIED BY:
1 !
1 !       V03-002 CWH3002         CW Hobbs                        4-Oct-1983
1 !               Change CTG to CBT so that a segmented SYS does not have to
1 !               be contiguous.
1 !
1 !       V03-001 CWH3001         CW Hobbs                        8-Sep-1983
1 !               Add the ability to segment a file, specifically so that
1 !               SYS.EXE can be split across two volumes.
1 !
1 !**
```

```
;    58           0057  1 LIBRARY 'SYS$LIBRARY:LIB';
;    59           0058  1
;    60           0059  1
;    61           0060  1 LITERAL
;    62           0061  1         TRUE=           1,
;    63           0062  1         FALSE=          0;
;    64           0063  1
;    65           0064  1
;    66           0065  1 STRUCTURE
;    67           0066  1         BBLOCK[O,P,S,E;N]=
;    68           0067  1             [N]
;    69           0068  1             (BBLOCK + O)<P,S,E>;
;    70           0069  1
;    71           0070  1
;    72           0071  1 PSECT
;    73           0072  1         CODE=           CODE,
;    74           0073  1         PLIT=           CODE,
;    75           0074  1         OWN=            DATA,
;    76           0075  1         GLOBAL=         DATA;
;    77           0076  1
;    78           0077  1
;    79           0078  1 FORWARD ROUTINE
;    80           0079  1         STABACCOP,
;    81           0080  1         FILE_ERROR:     NOVALUE;
;    82           0081  1
;    83           0082  1
;    84           0083  1 EXTERNAL ROUTINE
;    85           0084  1         LIB$GET_FOREIGN:        ADDRESSING_MODE(GENERAL),
;    86           0085  1         OTS$CVT_TI_L:           ADDRESSING_MODE(GENERAL);
;    88         0086   1 ROUTINE STABACCOP=
;    89         0087   1
;    90         0088   1 !++
;    91         0089   1 !
;    92         0090   1 ! FUNCTIONAL DESCRIPTION:
;    93         0091   1 !       This routine is the main entry point for the STABACCOP program.  It
;    94         0092   1 !       copies an image file, removing the appended patch text.  It is used and
;    95         0093   1 !       supported only for generation of Standalone BACKUP kits.
;    96         0094   1 !
;    97         0095   1 ! INPUT PARAMETERS:
;    98         0096   1 !       Standard VMS activation parameters (not used).
;    99         0097   1 !
;   100         0098   1 ! IMPLICIT INPUTS:
;   101         0099   1 !       NONE
;   102         0100   1 !
;   103         0101   1 ! OUTPUT PARAMETERS:
;   104         0102   1 !       NONE
;   105         0103   1 !
;   106         0104   1 ! IMPLICIT OUTPUTS:
;   107         0105   1 !       NONE
;   108         0106   1 !
;   109         0107   1 ! ROUTINE VALUE:
;   110         0108   1 !       Completion status.
;   111         0109   1 !
;   112         0110   1 ! SIDE EFFECTS:
;   113         0111   1 !       NONE
;   114         0112   1 !
;   115         0113   1 !--
;   116         0114   1
;   117         0115   2 BEGIN
;   118         0116   2 LOCAL
;   119         0117   2         COMMAND_DESC:   BBLOCK[DSC$K_S_BLN],     ! Descriptor for command buffer
;   120         0118   2         INFILE_DESC:    BBLOCK[DSC$K_S_BLN],     ! Descriptor for input file name
;   121         0119   2         OUTFILE_DESC:   BBLOCK[DSC$K_S_BLN],     ! Descriptor for output file name
;   122         0120   2         VAL_DESC:       BBLOCK[DSC$K_S_BLN],     ! Descriptor for numeric value
;   123         0121   2         COMMAND_BUFFER: VECTOR[132,BYTE],        ! Command buffer
;   124         0122   2         P,                                       ! Pointer to space
;   125         0123   2         INPUT_FAB:      $FAB_DECL,               ! FAB for input file
;   126         0124   2         INPUT_NAM:      $NAM_DECL,               ! NAM block for input file
;   127         0125   2         INPUT_XAB:      $XABFHC_DECL,            ! File header XAB for input file
;   128         0126   2         INPUT_RSA:      VECTOR[NAM$C_MAXRSS,BYTE], ! Resultant string for input file
;   129         0127   2         OUTPUT_FAB:     $FAB_DECL,               ! FAB for output file
;   130         0128   2         OUTPUT_RAB:     $RAB_DECL,               ! RAB for output file
;   131         0129   2         OUTPUT_NAM:     $NAM_DECL,               ! NAM block for output file
;   132         0130   2         OUTPUT_RSA:     VECTOR[NAM$C_MAXRSS,BYTE], ! Resultant string for output file
;   133         0131   2         RETADR:         VECTOR[2],               ! Return addresses from $CRMPSC
;   134         0132   2         IHD:            REF BBLOCK,              ! Pointer to IHD
;   135         0133   2         IHP:            REF BBLOCK,              ! Pointer to IHP
;   136         0134   2         FILE_SIZE,                               ! Size of file copied
;   137         0135   2         BLOCKS_LEFT,                             ! Number of blocks left to copy
;   138         0136   2         START_BLK,                               ! Starting block (numbered 0 to N-1)
;   139         0137   2         SEGMENT_SIZE    : INITIAL (99999),       ! Size of file segment
;   140         0138   2         INPUT_RSA_DESC: VECTOR[2],               ! Descriptor for input RSA
;   141         0139   2         OUTPUT_RSA_DESC:VECTOR[2],               ! Descriptor for output RSA
;   142         0140   2         STATUS_1,                                ! Status return
;   143         0141   2         STATUS_2;                                ! Status return
;   144         0142   2 LITERAL
```

```
2          FACILITY = 103;                                  ! Steal COPY's prefix
2
2
2   ! Get the foreign command line.  It must be of the form:
2   !          <input-file-specification> <space> <output-file-specification> [ <space> <start-vbn> <space> <segmen
2   !
2   COMMAND_DESC[DSC$W_LENGTH] = %ALLOCATION(COMMAND_BUFFER);
2   COMMAND_DESC[DSC$B_DTYPE] = DSC$K_DTYPE_T;
2   COMMAND_DESC[DSC$B_CLASS] = DSC$K_CLASS_S;
2   COMMAND_DESC[DSC$A_POINTER] = COMMAND_BUFFER;
2   STATUS_1 = LIB$GET_FOREIGN(COMMAND_DESC, 0, COMMAND_DESC);
2   IF NOT .STATUS_1 THEN RETURN .STATUS_1;
2
2
2   ! Locate the space between the input and output file specifications.
2   !
2   P = CH$FIND_CH(.COMMAND_DESC[DSC$W_LENGTH], .COMMAND_DESC[DSC$A_POINTER], %C' ');
2   IF .P EQL 0 THEN RETURN SS$_BADPARAM;
2   INFILE_DESC [DSC$W_LENGTH] = .P - .COMMAND_DESC [DSC$A_POINTER];          ! Set input name
2   INFILE_DESC [DSC$A_POINTER] = .COMMAND_DESC [DSC$A_POINTER];
2   COMMAND_DESC [DSC$A_POINTER] = .P + 1;                                  ! Remove input name
2   COMMAND_DESC [DSC$W_LENGTH] = .COMMAND_DESC [DSC$W_LENGTH] - 1 - .INFILE_DESC [DSC$W_LENGTH];
2
2   ! Locate the output file spec
2   !
2   P = CH$FIND_CH(.COMMAND_DESC[DSC$W_LENGTH], .COMMAND_DESC[DSC$A_POINTER], %C' ');
2   IF .P EQL 0
2   THEN
3       BEGIN
3       !
3       ! No optional numbers, the rest of the command string is the output file name
3       !
3       OUTFILE_DESC [DSC$W_LENGTH] = .COMMAND_DESC [DSC$W_LENGTH];
3       OUTFILE_DESC [DSC$A_POINTER] = .COMMAND_DESC [DSC$A_POINTER];
3       END
2   ELSE
3       BEGIN
3       OUTFILE_DESC [DSC$W_LENGTH] = .P - .COMMAND_DESC [DSC$A_POINTER];   ! Set input name
3       OUTFILE_DESC [DSC$A_POINTER] = .COMMAND_DESC [DSC$A_POINTER];
3       COMMAND_DESC [DSC$A_POINTER] = .P + 1;                                  ! Remove input name
3       COMMAND_DESC [DSC$W_LENGTH] = .COMMAND_DESC [DSC$W_LENGTH] - 1 - .OUTFILE_DESC [DSC$W_LENGTH];
3       !
3       ! Now get the two optional numbers
3       !
3       P = CH$FIND_CH(.COMMAND_DESC[DSC$W_LENGTH], .COMMAND_DESC[DSC$A_POINTER], %C' ');
3       IF .P EQL 0
3       THEN
3           RETURN SS$_BADPARAM;
3       VAL_DESC [DSC$W_LENGTH] = .P - .COMMAND_DESC [DSC$A_POINTER];       ! Set input name
3       VAL_DESC [DSC$A_POINTER] = .COMMAND_DESC [DSC$A_POINTER];
3       COMMAND_DESC [DSC$A_POINTER] = .P + 1;                                  ! Remove input name
3       COMMAND_DESC [DSC$W_LENGTH] = .COMMAND_DESC [DSC$W_LENGTH] - 1 - .VAL_DESC [DSC$W_LENGTH];
4       IF NOT (STATUS_1 = OTS$CVT_TI_L (VAL_DESC, START_BLK))
3       THEN
3           RETURN .STATUS_1;
3       IF (START_BLK = .START_BLK-1) LSS 0
3       THEN
```

```
;   202          0200 3             RETURN SS$_BADPARAM;
;   203          0201 4         IF NOT (STATUS_1 = OTS$CVT_TI_L (COMMAND_DESC, SEGMENT_SIZE))
;   204          0202 3         THEN
;   205          0203 3             RETURN .STATUS_1;
;   206          0204 2         END;
;   207          0205 2
;   208          0206 2     ! Open the input file.
;   209          0207 2     !
;   210        P 0208 2     $FAB_INIT(FAB=INPUT_FAB,
;   211        P 0209 2         DNA=UPLIT BYTE('.EXE'),
;   212        P 0210 2         DNS=%CHARCOUNT('.EXE'),
;   213        P 0211 2         FNA=.INFILE_DESC[DSC$A_POINTER],
;   214        P 0212 2         FNS=.INFILE_DESC[DSC$W_LENGTH],
;   215        P 0213 2         FOP=UFO,
;   216        P 0214 2         NAM=INPUT_NAM,
;   217          0215 2         XAB=INPUT_XAB);
;   218        P 0216 2     $NAM_INIT(NAM=INPUT_NAM,
;   219        P 0217 2         RSA=INPUT_RSA,
;   220          0218 2         RSS=NAM$C_MAXRSS);
;   221          0219 2     $XABFHC_INIT(XAB=INPUT_XAB);
;   222          0220 3     IF NOT $OPEN(FAB=INPUT_FAB)
;   223          0221 2     THEN
;   224          0222 2         FILE_ERROR(FACILITY^16 + SHR$_OPENIN + STS$K_SEVERE, INPUT_FAB, INPUT_FAB);
;   225          0223 2
;   226          0224 2
;   227          0225 2     ! Ensure that the file has appropriate characteristics for an image file:
;   228          0226 2     ! sequential with fixed length 512 byte records and no record attributes.
;   229          0227 2     !
;   230          0228 2     IF .INPUT_FAB[FAB$B_ORG] NEQ FAB$C_SEQ
;   231          0229 2     OR .INPUT_FAB[FAB$B_RFM] NEQ FAB$C_FIX
;   232          0230 2     OR .INPUT_FAB[FAB$W_MRS] NEQ 512
;   233          0231 2     OR .INPUT_FAB[FAB$B_RAT] NEQ 0
;   234          0232 2     THEN
;   235          0233 3         BEGIN
;   236          0234 3         INPUT_FAB[FAB$L_STS] = SS$_BADIMGHDR;
;   237          0235 3         INPUT_FAB[FAB$L_STV] = 0;
;   238          0236 3         FILE_ERROR(FACILITY^16 + SHR$_OPENIN + STS$K_SEVERE, INPUT_FAB, INPUT_FAB);
;   239          0237 2         END;
;   240          0238 2
;   241          0239 2
;   242          0240 2     ! Map the file into memory.
;   243          0241 2     !
;   244        P 0242 2     STATUS_2 = $CRMPSC(
;   245        P 0243 2         INADR=UPLIT(0, 0),
;   246        P 0244 2         RETADR=RETADR,
;   247        P 0245 2         FLAGS=SEC$M_CRF OR SEC$M_EXPREG OR SEC$M_WRT,
;   248          0246 2         CHAN=.INPUT_FAB[FAB$L_STV]);
;   249          0247 2     IF NOT .STATUS_2
;   250          0248 2     THEN
;   251          0249 3         BEGIN
;   252          0250 3         INPUT_FAB[FAB$L_STS] = .STATUS_2;
;   253          0251 3         INPUT_FAB[FAB$L_STV] = 0;
;   254          0252 3         FILE_ERROR(FACILITY^16 + SHR$_OPENIN + STS$K_SEVERE, INPUT_FAB, INPUT_FAB);
;   255          0253 2         END;
;   256          0254 2
;   257          0255 2
;   258          0256 2     ! Examine the image header to determine the location of the patch text.
;   259             0257  2 !
;   260             0258  2 IHD = .RETADR[0];
;   261             0259  2 IF .IHD[IHD$W_PATCHOFF] NEQ 0
;   262             0260  2 THEN
;   263             0261  3     BEGIN
;   264             0262  3     IHP = .IHD + .IHD[IHD$W_PATCHOFF];
;   265             0263  3     IF .IHP[IHP$L_PATCOMTXT] NEQ 0
;   266             0264  3     THEN
;   267             0265  4         BEGIN
;   268             0266  4         INPUT_XAB[XAB$L_EBK] = .IHP[IHP$L_PATCOMTXT];
;   269             0267  4         INPUT_XAB[XAB$W_FFB] = 0;
;   270             0268  4         IHP[IHP$L_PATCOMTXT] = 0;
;   271             0269  3         END;
;   272             0270  2     END;
;   273             0271  2
;   274             0272  2 ! Determine the size of the output file
;   275             0273  2 !
;   276             0274  2 IF .INPUT_XAB[XAB$W_FFB] EQL 0
;   277             0275  2 THEN
;   278             0276  2     INPUT_XAB[XAB$L_EBK] = .INPUT_XAB[XAB$L_EBK] - 1;
;   279             0277  2 FILE_SIZE = MIN ((.INPUT_XAB[XAB$L_EBK]-.START_BLK), .SEGMENT_SIZE);
;   280             0278  2
;   281             0279  2 ! Create the output file.
;   282             0280  2 !
;   283           P 0281  2 $FAB_INIT(FAB=OUTPUT_FAB,
;   284           P 0282  2     ALQ=.FILE_SIZE,
;   285           P 0283  2     DNA=UPLIT BYTE('.EXE'),
;   286           P 0284  2     DNS=%CHARCOUNT('.EXE'),
;   287           P 0285  2     FAC=BIO,
;   288           P 0286  2     FNA=.OUTFILE_DESC[DSC$A_POINTER],
;   289           P 0287  2     FNS=.OUTFILE_DESC[DSC$W_LENGTH],
;   290           P 0288  2     FOP=<CTG,OFP>,
;   291           P 0289  2     MRS=512,
;   292           P 0290  2     NAM=OUTPUT_NAM,
;   293           P 0291  2     ORG=SEQ,
;   294             0292  2     RFM=FIX);
;   295           P 0293  2 $RAB_INIT(RAB=OUTPUT_RAB,
;   296           P 0294  2     FAB=OUTPUT_FAB,
;   297             0295  2     ROP=BIO);
;   298           P 0296  2 $NAM_INIT(NAM=OUTPUT_NAM,
;   299           P 0297  2     RLF=INPUT_NAM,
;   300           P 0298  2     RSA=OUTPUT_RSA,
;   301             0299  2     RSS=NAM$C_MAXRSS);
;   302             0300  2 IF .SEGMENT_SIZE NEQ 99999       ! If segmenting, then don't need contiguous
;   303             0301  2 THEN
;   304             0302  3     BEGIN
;   305             0303  3     OUTPUT_FAB [FAB$V_CTG] = FALSE;
;   306             0304  3     OUTPUT_FAB [FAB$V_CBT] = TRUE;
;   307             0305  2     END;
;   308             0306  2 IF NOT $CREATE(FAB=OUTPUT_FAB)
;   309             0307  2 THEN
;   310             0308  2     FILE_ERROR(FACILITY^16 + SHR$_OPENOUT + STS$K_SEVERE, OUTPUT_FAB, OUTPUT_FAB);
;   311             0309  2 IF NOT $CONNECT(RAB=OUTPUT_RAB)
;   312             0310  2 THEN
;   313             0311  2     FILE_ERROR(FACILITY^16 + SHR$_OPENOUT + STS$K_SEVERE, OUTPUT_FAB, OUTPUT_RAB);
;   314             0312  2
;   315             0313  2
;    316          0314  2 ! Write the output file.
;    317          0315  2 !
;    318          0316  2 OUTPUT_RAB[RAB$L_RBF] = (.IHD + (.START_BLK*512));
;    319          0317  2 BLOCKS_LEFT = .FILE_SIZE;
;    320          0318  2 WHILE .BLOCKS_LEFT GTR 0 DO
;    321          0319  3     BEGIN
;    322          0320  3     LOCAL
;    323          0321  3         BLOCKS;
;    324          0322  3
;    325          0323  3     BLOCKS = MIN(.BLOCKS_LEFT, 127);
;    326          0324  3     BLOCKS_LEFT = .BLOCKS_LEFT - .BLOCKS;
;    327          0325  3     OUTPUT_RAB[RAB$W_RSZ] = .BLOCKS * 512;
;    328          0326  4     IF NOT $WRITE(RAB=OUTPUT_RAB)
;    329          0327  3     THEN
;    330          0328  3         FILE_ERROR(FACILITY^16 + SHR$_WRITEERR + STS$K_SEVERE, OUTPUT_FAB, OUTPUT_RAB);
;    331          0329  3     OUTPUT_RAB[RAB$L_RBF] = .OUTPUT_RAB[RAB$L_RBF] + .OUTPUT_RAB[RAB$W_RSZ];
;    332          0330  2     END;
;    333          0331  2
;    334          0332  2
;    335          0333  2 ! Close the output file.
;    336          0334  2 !
;    337          0335  3 IF NOT $CLOSE(FAB=OUTPUT_FAB)
;    338          0336  2 THEN
;    339          0337  2     FILE_ERROR(FACILITY^16 + SHR$_CLOSEOUT + STS$K_SEVERE, OUTPUT_FAB, OUTPUT_FAB);
;    340          0338  2
;    341          0339  2
;    342          0340  2 ! Log the copied message.
;    343          0341  2 !
;    344          0342  2 INPUT_RSA_DESC[0] = .INPUT_NAM[NAM$B_RSL];
;    345          0343  2 INPUT_RSA_DESC[1] = .INPUT_NAM[NAM$L_RSA];
;    346          0344  2 OUTPUT_RSA_DESC[0] = .OUTPUT_NAM[NAM$B_RSL];
;    347          0345  2 OUTPUT_RSA_DESC[1] = .OUTPUT_NAM[NAM$L_RSA];
;    348          0346  2 SIGNAL(FACILITY^16 + SHR$_COPIEDB + STS$K_SUCCESS, 3, INPUT_RSA_DESC, OUTPUT_RSA_DESC, .FILE_SIZE);
;    349          0347  2
;    350          0348  2
;    351          0349  2 ! Return with success.
;    352          0350  2 !
;    353          0351  2 SS$_NORMAL
;    354          0352  1 END;


                                                                         .TITLE  STABACCOP Copy image file for Standalone BACKUP
;                                                                                                  kit
                                                                         .IDENT  \V04-000\

                                                                         .PSECT  CODE,NOWRT,2

                                      45  58  45  2E  00000 P.AAA:       .ASCII  \.EXE\                                             ;
                                00000000  00000000  00004 P.AAB:         .LONG   0, 0                                               ;
                                      45  58  45  2E  0000C P.AAC:       .ASCII  \.EXE\                                             ;

                                                                         .EXTRN  LIB$GET_FOREIGN
                                                                         .EXTRN  OTS$CVT_TI_L, SYS$OPEN
                                                                         .EXTRN  SYS$CRMPSC, SYS$CREATE
                                                                         .EXTRN  SYS$CONNECT, SYS$WRITE
                                                                         .EXTRN  SYS$CLOSE
                                      03FC 00000 STABACCOP:
                                                       .WORD    Save R2,R3,R4,R5,R6,R7,R8,R9            ; 0086
                    59   00000000G  00  9E 00002       MOVAB    OTS$CVT_TI_L, R9                        ;
                    58       0000V  CF  9E 00009       MOVAB    FILE_ERROR, R8                          ;
                    5E        FB6C  CE  9E 0000E       MOVAB    -1172(SP), SP                           ;
              04    AE    0001869F  8F  D0 00013       MOVL     #99999, SEGMENT_SIZE                    ; 0115
              F8    AD    010E0084  8F  D0 0001B       MOVL     #17694852, COMMAND_DESC                 ; 0149
              FC    AD        FF5C  CD  9E 00023       MOVAB    COMMAND_BUFFER, COMMAND_DESC+4          ; 0152
                                F8  AD  9F 00029       PUSHAB   COMMAND_DESC                            ; 0153
                                    7E  D4 0002C       CLRL     -(SP)                                   ;
                                F8  AD  9F 0002E       PUSHAB   COMMAND_DESC                            ;
        00000000G   00              03  FB 00031       CALLS    #3, LIB$GET_FOREIGN                     ;
                    52              50  D0 00038       MOVL     R0, STATUS_1                            ;
                    03              52  E8 0003B       BLBS     STATUS_1, 1$                            ; 0154
                                  00BA  31 0003E       BRW      9$                                      ;
FC    BD      F8    AD              20  3A 00041 1$:   LOCC     #32, COMMAND_DESC, @COMMAND_DESC+4      ; 0159
                                    02  12 00047       BNEQ     2$                                      ;
                                    51  D4 00049       CLRL     R1                                      ;
                                    51  D5 0004B 2$:   TSTL     P                                       ; 0160
                                    66  13 0004D       BEQL     6$                                      ;
F0    AD            51          FC  AD  A3 0004F       SUBW3    COMMAND_DESC+4, P, INFILE_DESC          ; 0161
              F4    AD          FC  AD  D0 00055       MOVL     COMMAND_DESC+4, INFILE_DESC+4           ; 0162
              FC    AD          01  A1  9E 0005A       MOVAB    1(R1), COMMAND_DESC+4                   ; 0163
                    50          F8  AD  3C 0005F       MOVZWL   COMMAND_DESC, R0                        ; 0164
                    53          F0  AD  3C 00063       MOVZWL   INFILE_DESC, R3                         ;
                    50              53  C2 00067       SUBL2    R3, R0                                  ;
F8    AD            50              01  A3 0006A       SUBW3    #1, R0, COMMAND_DESC                    ;
FC    BD      F8    AD              20  3A 0006F       LOCC     #32, COMMAND_DESC, @COMMAND_DESC+4      ; 0168
                                    02  12 00075       BNEQ     3$                                      ;
                                    51  D4 00077       CLRL     R1                                      ;
                                    51  D5 00079 3$:   TSTL     P                                       ; 0169
                                    0C  12 0007B       BNEQ     4$                                      ;
              E8    AD          F8  AD  B0 0007D       MOVW     COMMAND_DESC, OUTFILE_DESC              ; 0175
              EC    AD          FC  AD  D0 00082       MOVL     COMMAND_DESC+4, OUTFILE_DESC+4          ; 0176
                                    76  11 00087       BRB      10$                                     ; 0169
E8    AD            51          FC  AD  A3 00089 4$:   SUBW3    COMMAND_DESC+4, P, OUTFILE_DESC         ; 0180
              EC    AD          FC  AD  D0 0008F       MOVL     COMMAND_DESC+4, OUTFILE_DESC+4          ; 0181
              FC    AD          01  A1  9E 00094       MOVAB    1(R1), COMMAND_DESC+4                   ; 0182
                    50          F8  AD  3C 00099       MOVZWL   COMMAND_DESC, R0                        ; 0183
                    53          E8  AD  3C 0009D       MOVZWL   OUTFILE_DESC, R3                        ;
                    50              53  C2 000A1       SUBL2    R3, R0                                  ;
F8    AD            50              01  A3 000A4       SUBW3    #1, R0, COMMAND_DESC                    ;
FC    BD      F8    AD              20  3A 000A9       LOCC     #32, COMMAND_DESC, @COMMAND_DESC+4      ; 0187
                                    02  12 000AF       BNEQ     5$                                      ;
                                    51  D4 000B1       CLRL     R1                                      ;
                                    51  D5 000B3 5$:   TSTL     P                                       ; 0188
                                    31  13 000B5 6$:   BEQL     7$                                      ;
E0    AD            51          FC  AD  A3 000B7       SUBW3    COMMAND_DESC+4, P, VAL_DESC             ; 0191
              E4    AD          FC  AD  D0 000BD       MOVL     COMMAND_DESC+4, VAL_DESC+4              ; 0192
              FC    AD          01  A1  9E 000C2       MOVAB    1(R1), COMMAND_DESC+4                   ; 0193
                    50          F8  AD  3C 000C7       MOVZWL   COMMAND_DESC, R0                        ; 0194
                    51          E0  AD  3C 000CB       MOVZWL   VAL_DESC, R1                            ;
                    50              51  C2 000CF       SUBL2    R1, R0                                  ;
F8    AD            50              01  A3 000D2       SUBW3    #1, R0, COMMAND_DESC                    ;
                                    5E  DD 000D7       PUSHL    SP                                      ; 0195
                                E0  AD  9F 000D9       PUSHAB   VAL_DESC                                ;
                    69              02  FB 000DC       CALLS    #2, OTS$CVT_TI_L                        ;
```

```
                            52           50 D0 000DF        MOVL    R0, STATUS_1                    ;
                            16           52 E9 000E2        BLBC    STATUS_1, 9$                    ;
                            04           6E F4 000E5        SOBGEQ  START_BLK, 8$                   ; 0198
                            50           14 D0 000E8 7$:    MOVL    #20, R0                         ; 0200
                                            04 000EB        RET                                     ;
                                     04  AE 9F 000EC 8$:    PUSHAB  SEGMENT_SIZE                    ; 0201
                                     F8  AD 9F 000EF        PUSHAB  COMMAND_DESC                    ;
                            69           02 FB 000F2        CALLS   #2, OTS$CVT_TI_L                ;
                            52           50 D0 000F5        MOVL    R0, STATUS_1                    ;
                            04           52 E8 000F8        BLBS    STATUS_1, 10$                   ;
                            50           52 D0 000FB 9$:    MOVL    STATUS_1, R0                    ; 0203
                                            04 000FE        RET                                     ;
  0050  8F       00         6E           00 2C 000FF 10$:   MOVC5   #0, (SP), #0, #80, $RMS_PTR     ; 0215
                                   FF0C  CD    00106                                                ;
                       FF0C CD     5003  8F B0 00109        MOVW    #20483, $RMS_PTR                ;
                       FF10 CD 00020000  8F D0 00110        MOVL    #131072, $RMS_PTR+4             ;
                       FF22 CD           02 90 00119        MOVB    #2, $RMS_PTR+22                 ;
                       FF2B CD           02 90 0011E        MOVB    #2, $RMS_PTR+31                 ;
                       FF30 CD     FE80  CD 9E 00123        MOVAB   INPUT_XAB, $RMS_PTR+36          ;
                       FF34 CD     FEAC  CD 9E 0012A        MOVAB   INPUT_NAM, $RMS_PTR+40          ;
                       FF38 CD       F4  AD D0 00131        MOVL    INFILE_DESC+4, $RMS_PTR+44      ;
                       FF3C CD     FEB5  CF 9E 00137        MOVAB   P.AAA, $RMS_PTR+48              ;
                       FF40 CD       F0  AD 90 0013E        MOVB    INFILE_DESC, $RMS_PTR+52        ;
                       FF41 CD           04 90 00144        MOVB    #4, $RMS_PTR+53                 ;
  0060  8F       00         6E           00 2C 00149        MOVC5   #0, (SP), #0, #96, $RMS_PTR     ; 0218
                                   FEAC  CD    00150                                                ;
                       FEAC CD     6002  8F B0 00153        MOVW    #24578, $RMS_PTR                ;
                       FEAE CD           01 8E 0015A        MNEGB   #1, $RMS_PTR+2                  ;
                       FEB0 CD     0214  CE 9E 0015F        MOVAB   INPUT_RSA, $RMS_PTR+4           ;
        2C       00         6E           00 2C 00166        MOVC5   #0, (SP), #0, #44, $RMS_PTR     ; 0219
                                   FE80  CD    0016B                                                ;
                       FE80 CD     2C1D  8F B0 0016E        MOVW    #11293, $RMS_PTR                ;
                                   FF0C  CD 9F 00175        PUSHAB  INPUT_FAB                       ; 0220
                  00000000G 00           01 FB 00179        CALLS   #1, SYS$OPEN                    ;
                            11           50 E8 00180        BLBS    R0, 11$                         ;
                                   FF0C  CD 9F 00183        PUSHAB  INPUT_FAB                       ; 0222
                                   FF0C  CD 9F 00187        PUSHAB  INPUT_FAB                       ;
                               0067109C  8F DD 0018B        PUSHL   #6754460                        ;
                            68           03 FB 00191        CALLS   #3, FILE_ERROR                  ;
                                   FF29  CD 95 00194 11$:   TSTB    INPUT_FAB+29                    ; 0228
                                         16 12 00198        BNEQ    12$                             ;
                            01     FF2B  CD 91 0019A        CMPB    INPUT_FAB+31, #1                ; 0229
                                         0F 12 0019F        BNEQ    12$                             ;
                       0200 8F     FF42  CD B1 001A1        CMPW    INPUT_FAB+54, #512              ; 0230
                                         06 12 001A8        BNEQ    12$                             ;
                                   FF2A  CD 95 001AA        TSTB    INPUT_FAB+30                    ; 0231
                                         1B 13 001AE        BEQL    13$                             ;
                       FF14 CD       44  8F 9A 001B0 12$:   MOVZBL  #68, INPUT_FAB+8                ; 0234
                                   FF18  CD D4 001B6        CLRL    INPUT_FAB+12                    ; 0235
                                   FF0C  CD 9F 001BA        PUSHAB  INPUT_FAB                       ; 0236
                                   FF0C  CD 9F 001BE        PUSHAB  INPUT_FAB                       ;
                               0067109C  8F DD 001C2        PUSHL   #6754460                        ;
                            68           03 FB 001C8        CALLS   #3, FILE_ERROR                  ;
                                         7E 7C 001CB 13$:   CLRQ    -(SP)                           ; 0246
                                         7E 7C 001CD        CLRQ    -(SP)                           ;
                                   FF18  CD DD 001CF        PUSHL   INPUT_FAB+12                    ;
                                         7E 7C 001D3        CLRQ    -(SP)                           ;
```

```
                                                           7E  D4  001D5         CLRL    -(SP)                               ;
                                                 0002000A  8F  DD  001D7         PUSHL   #131082                             ;
                                                           7E  D4  001DD         CLRL    -(SP)                               ;
                                                       40  AE  9F  001DF         PUSHAB  RETADR                              ;
                                                     FE0E  CF  9F  001E2         PUSHAB  P.AAB                               ;
                                00000000G   00             0C  FB  001E6         CALLS   #12, SYS$CRMPSC                     ;
                                            1A             50  E8  001ED         BLBS    STATUS_2, 14$                       ; 0247
                                     FF14   CD             50  D0  001F0         MOVL    STATUS_2, INPUT_FAB+8               ; 0250
                                                     FF18  CD  D4  001F5         CLRL    INPUT_FAB+12                        ; 0251
                                                     FF0C  CD  9F  001F9         PUSHAB  INPUT_FAB                           ; 0252
                                                     FF0C  CD  9F  001FD         PUSHAB  INPUT_FAB                           ;
                                                 0067109C  8F  DD  00201         PUSHL   #6754460                            ;
                                            68             03  FB  00207         CALLS   #3, FILE_ERROR                      ;
                                            56         18  AE  D0  0020A 14$:    MOVL    RETADR, IHD                         ; 0258
                                                       08  A6  B5  0020E         TSTW    8(IHD)                              ; 0259
                                                           19  13  00211         BEQL    15$                                 ;
                                            50         08  A6  3C  00213         MOVZWL  8(IHD), IHP                         ; 0262
                                            50             56  C0  00217         ADDL2   IHD, IHP                            ;
                                                       20  A0  D5  0021A         TSTL    32(IHP)                             ; 0263
                                                           0D  13  0021D         BEQL    15$                                 ;
                                     FE90   CD         20  A0  D0  0021F         MOVL    32(IHP), INPUT_XAB+16               ; 0266
                                                     FE94  CD  B4  00225         CLRW    INPUT_XAB+20                        ; 0267
                                                       20  A0  D4  00229         CLRL    32(IHP)                             ; 0268
                                                     FE94  CD  B5  0022C 15$:    TSTW    INPUT_XAB+20                        ; 0274
                                                           04  12  00230         BNEQ    16$                                 ;
                                                     FE90  CD  D7  00232         DECL    INPUT_XAB+16                        ; 0276
                              50     FE90   CD             6E  C3  00236 16$:    SUBL3   START_BLK, INPUT_XAB+16, R0         ; 0277
                                       04   AE             50  D1  0023C         CMPL    R0, SEGMENT_SIZE                    ;
                                                           04  15  00240         BLEQ    17$                                 ;
                                            50         04  AE  D0  00242         MOVL    SEGMENT_SIZE, R0                    ;
                                            57             50  D0  00246 17$:    MOVL    R0, FILE_SIZE                       ;
      0050    8F              00            6E             00  2C  00249         MOVC5   #0, (SP), #0, #80, $RMS_PTR         ; 0292
                                                     01C4  CE      00250                                                     ;
                                     01C4   CE       5003  8F  B0  00253         MOVW    #20483, $RMS_PTR                    ;
                                     01C8   CE   20100000  8F  D0  0025A         MOVL    #537919488, $RMS_PTR+4              ;
                                     01D4   CE             57  D0  00263         MOVL    FILE_SIZE, $RMS_PTR+16              ;
                                     01DA   CE             20  90  00268         MOVB    #32, $RMS_PTR+22                    ;
                                                     01E1  CE  94  0026D         CLRB    $RMS_PTR+29                         ;
                                     01E3   CE             01  90  00271         MOVB    #1, $RMS_PTR+31                     ;
                                     01EC   CE       0120  CE  9E  00276         MOVAB   OUTPUT_NAM, $RMS_PTR+40             ;
                                     01F0   CE         EC  AD  D0  0027D         MOVL    OUTFILE_DESC+4, $RMS_PTR+44         ;
                                     01F4   CE       FD75  CF  9E  00283         MOVAB   P.AAC, $RMS_PTR+48                  ;
                                     01F8   CE         E8  AD  90  0028A         MOVB    OUTFILE_DESC, $RMS_PTR+52           ;
                                     01F9   CE             04  90  00290         MOVB    #4, $RMS_PTR+53                     ;
                                     01FA   CE       0200  8F  B0  00295         MOVW    #512, $RMS_PTR+54                   ;
      0044    8F              00            6E             00  2C  0029C         MOVC5   #0, (SP), #0, #68, $RMS_PTR         ; 0295
                                                     0180  CE      002A3                                                     ;
                                     0180   CE       4401  8F  B0  002A6         MOVW    #17409, $RMS_PTR                    ;
                                     0184   CE       0800  8F  3C  002AD         MOVZWL  #2048, $RMS_PTR+4                   ;
                                     01BC   CE       01C4  CE  9E  002B4         MOVAB   OUTPUT_FAB, $RMS_PTR+60             ;
      0060    8F              00            6E             00  2C  002BB         MOVC5   #0, (SP), #0, #96, $RMS_PTR         ; 0299
                                                     0120  CE      002C2                                                     ;
                                     0120   CE       6002  8F  B0  002C5         MOVW    #24578, $RMS_PTR                    ;
                                     0122   CE             01  8E  002CC         MNEGB   #1, $RMS_PTR+2                      ;
                                     0124   CE         20  AE  9E  002D1         MOVAB   OUTPUT_RSA, $RMS_PTR+4              ;
                                     0130   CE       FEAC  CD  9E  002D7         MOVAB   INPUT_NAM, $RMS_PTR+16              ;
                                 0001869F   8F         04  AE  D1  002DE         CMPL    SEGMENT_SIZE, #99999                ; 0300
```

```
                                           0A  13 002E6          BEQL    18$                                      ;
                   01CA   CE               10  8A 002E8          BICB2   #16, OUTPUT_FAB+6                        ; 0303
                   01CA   CE               20  88 002ED          BISB2   #32, OUTPUT_FAB+6                        ; 0304
                                 01C4      CE  9F 002F2  18$:    PUSHAB  OUTPUT_FAB                               ; 0306
              00000000G   00               01  FB 002F6          CALLS   #1, SYS$CREATE                           ;
                          11               50  E8 002FD          BLBS    R0, 19$                                  ;
                                 01C4      CE  9F 00300          PUSHAB  OUTPUT_FAB                               ; 0308
                                 01C8      CE  9F 00304          PUSHAB  OUTPUT_FAB                               ;
                             006710A4      8F  DD 00308          PUSHL   #6754468                                 ;
                          68               03  FB 0030E          CALLS   #3, FILE_ERROR                           ;
                                 0180      CE  9F 00311  19$:    PUSHAB  OUTPUT_RAB                               ; 0309
              00000000G   00               01  FB 00315          CALLS   #1, SYS$CONNECT                          ;
                          11               50  E8 0031C          BLBS    R0, 20$                                  ;
                                 0180      CE  9F 0031F          PUSHAB  OUTPUT_RAB                               ; 0311
                                 01C8      CE  9F 00323          PUSHAB  OUTPUT_FAB                               ;
                             006710A4      8F  DD 00327          PUSHL   #6754468                                 ;
                          68               03  FB 0032D          CALLS   #3, FILE_ERROR                           ;
             50           6E               09  78 00330  20$:    ASHL    #9, START_BLK, R0                        ; 0316
    01A8     CE           50               56  C1 00334          ADDL3   IHD, R0, OUTPUT_RAB+40                   ;
                          52               57  D0 0033A          MOVL    FILE_SIZE, BLOCKS_LEFT                   ; 0317
                                           52  D5 0033D  21$:    TSTL    BLOCKS_LEFT                              ; 0318
                                           46  15 0033F          BLEQ    24$                                      ;
                          50               52  D0 00341          MOVL    BLOCKS_LEFT, R0                          ; 0323
              0000007F    8F               50  D1 00344          CMPL    R0, #127                                 ;
                                           04  15 0034B          BLEQ    22$                                      ;
                          50         7F    8F  9A 0034D          MOVZBL  #127, R0                                 ;
                          52               50  C2 00351  22$:    SUBL2   BLOCKS, BLOCKS_LEFT                      ; 0324
    01A2     CE           50       0200    8F  A5 00354          MULW3   #512, BLOCKS, OUTPUT_RAB+34              ; 0325
                                 0180      CE  9F 0035C          PUSHAB  OUTPUT_RAB                               ; 0326
              00000000G   00               01  FB 00360          CALLS   #1, SYS$WRITE                            ;
                          11               50  E8 00367          BLBS    R0, 23$                                  ;
                                 0180      CE  9F 0036A          PUSHAB  OUTPUT_RAB                               ; 0328
                                 01C8      CE  9F 0036E          PUSHAB  OUTPUT_FAB                               ;
                             006710D4      8F  DD 00372          PUSHL   #6754516                                 ;
                          68               03  FB 00378          CALLS   #3, FILE_ERROR                           ;
                          50     01A2      CE  3C 0037B  23$:    MOVZWL  OUTPUT_RAB+34, R0                        ; 0329
                   01A8   CE               50  C0 00380          ADDL2   R0, OUTPUT_RAB+40                        ;
                                           B6  11 00385          BRB     21$                                      ; 0318
                                 01C4      CE  9F 00387  24$:    PUSHAB  OUTPUT_FAB                               ; 0335
              00000000G   00               01  FB 0038B          CALLS   #1, SYS$CLOSE                            ;
                          11               50  E8 00392          BLBS    R0, 25$                                  ;
                                 01C4      CE  9F 00395          PUSHAB  OUTPUT_FAB                               ; 0337
                                 01C8      CE  9F 00399          PUSHAB  OUTPUT_FAB                               ;
                             0067105C      8F  DD 0039D          PUSHL   #6754396                                 ;
                          68               03  FB 003A3          CALLS   #3, FILE_ERROR                           ;
                     10   AE     FEAF      CD  9A 003A6  25$:    MOVZBL  INPUT_NAM+3, INPUT_RSA_DESC              ; 0342
                     14   AE     FEB0      CD  D0 003AC          MOVL    INPUT_NAM+4, INPUT_RSA_DESC+4            ; 0343
                     08   AE     0123      CE  9A 003B2          MOVZBL  OUTPUT_NAM+3, OUTPUT_RSA_DESC            ; 0344
                     0C   AE     0124      CE  D0 003B8          MOVL    OUTPUT_NAM+4, OUTPUT_RSA_DESC+4          ; 0345
                                           57  DD 003BE          PUSHL   FILE_SIZE                                ; 0346
                                   0C      AE  9F 003C0          PUSHAB  OUTPUT_RSA_DESC                          ;
                                   18      AE  9F 003C3          PUSHAB  INPUT_RSA_DESC                           ;
                                           03  DD 003C6          PUSHL   #3                                       ;
                             00671061      8F  DD 003C8          PUSHL   #6754401                                 ;
              00000000G   00               05  FB 003CE          CALLS   #5, LIB$SIGNAL                           ;
                          50               01  D0 003D5          MOVL    #1, R0                                   ; 0352
                                               04 003D8          RET                                              ;
```

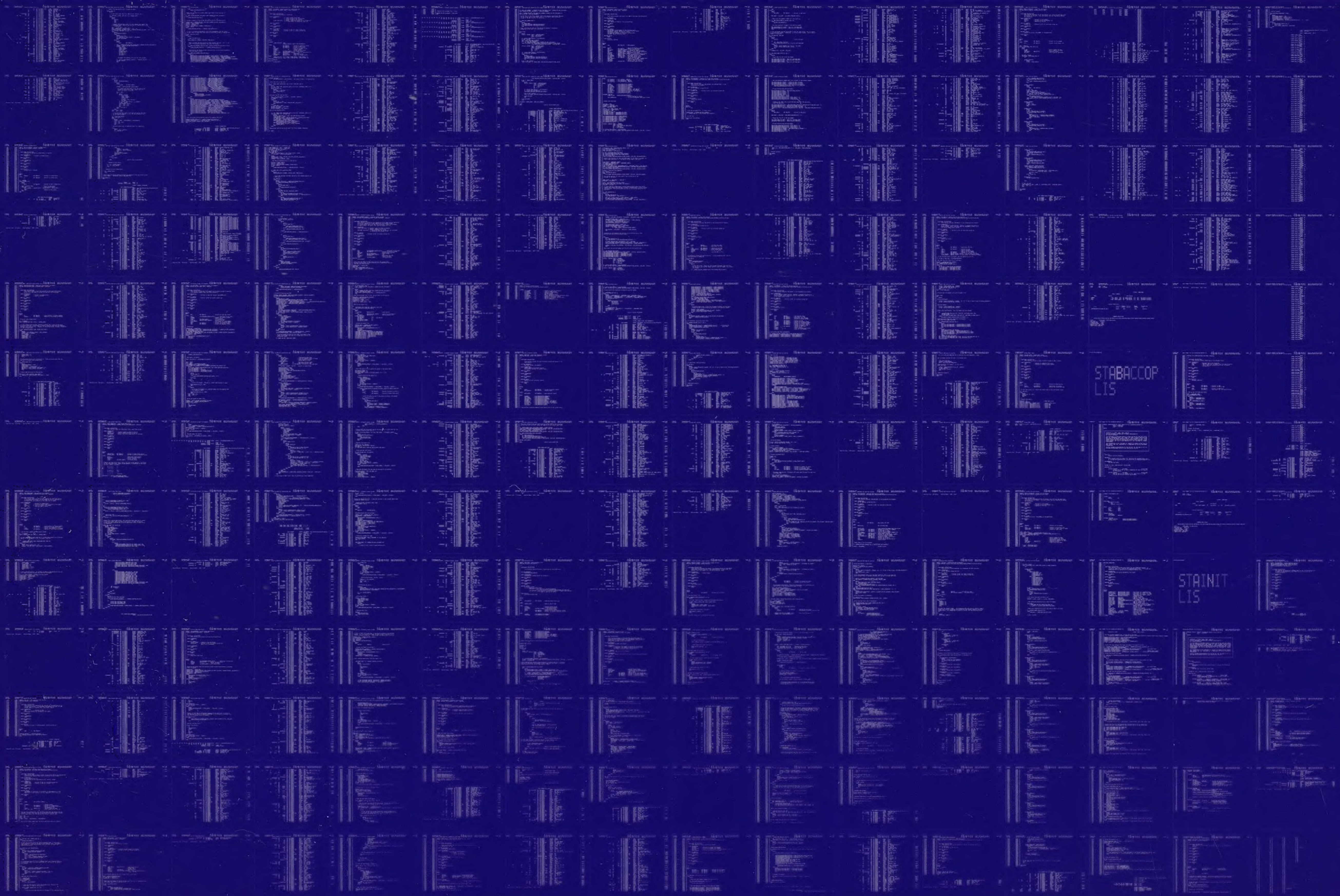
```
; Routine Size:  985 bytes,    Routine Base:  CODE + 0010
```

```
1 ROUTINE FILE_ERROR(MESSAGE,FAB,FRAB): NOVALUE=
1
1 !++
1 !
1 ! FUNCTIONAL DESCRIPTION:
1 !       This routine handles signalling of file-related errors.
1 !
1 ! INPUT PARAMETERS:
1 !       MESSAGE         - Message code to be signalled.
1 !       FAB             - FAB for the file, to obtain the file specification.
1 !       FRAB            - FAB or RAB that sustained the error, to obtain
1 !                         STS and STV.
1 !
1 ! IMPLICIT INPUTS:
1 !       NONE
1 !
1 ! OUTPUT PARAMETERS:
1 !       NONE
1 !
1 ! IMPLICIT OUTPUTS:
1 !       NONE
1 !
1 ! ROUTINE VALUE:
1 !       NONE
1 !
1 ! SIDE EFFECTS:
1 !       Message signalled.
1 !
1 !--
1
2 BEGIN
2 MAP
2       FAB:            REF BBLOCK,     ! Pointer to FAB
2       FRAB:           REF BBLOCK;     ! Pointer to FAB or RAB
2 LOCAL
2       NAM:            REF BBLOCK,     ! Pointer to NAM block
2       DESC:           VECTOR[2];      ! Descriptor for file specification
2
2
2 ! Set up the file name descriptor.
2 !
2 NAM = .FAB[FAB$L_NAM];
2 IF .NAM[NAM$B_RSL] NEQ 0
2 THEN
3     BEGIN
3     DESC[0] = .NAM[NAM$B_RSL];
3     DESC[1] = .NAM[NAM$L_RSA];
3     END
2 ELSE IF .NAM[NAM$B_ESL] NEQ 0
2 THEN
3     BEGIN
3     DESC[0] = .NAM[NAM$B_ESL];
3     DESC[1] = .NAM[NAM$L_ESA];
3     END
2 ELSE
3     BEGIN
3     DESC[0] = .FAB[FAB$B_FNS];
```

```
;  413          0410  3       DESC[1] = .FAB[FAB$L_FNA];
;  414          0411  2       END;
;  415          0412  2
;  416          0413  2
;  417          0414  2  ! Signal the message.
;  418          0415  2  !
;  419          0416  2  SIGNAL(.MESSAGE, 1, DESC, .FRAB[FAB$L_STS], .FRAB[FAB$L_STV]);
;  420          0417  1  END;



                                         0000 00000 FILE_ERROR:
                                                               .WORD    Save nothing                                      ; 0353
                     5E           08  C2 00002                 SUBL2    #8, SP                                            ;
                     51      08   AC  D0 00005                 MOVL     FAB, R1                                           ; 0394
                     50      28   A1  D0 00009                 MOVL     40(R1), NAM                                       ;
                             03   A0  95 0000D                 TSTB     3(NAM)                                            ; 0395
                                  0B  13 00010                 BEQL     1$                                                ;
                     6E      03   A0  9A 00012                 MOVZBL   3(NAM), DESC                                      ; 0398
              04     AE      04   A0  D0 00016                 MOVL     4(NAM), DESC+4                                    ; 0399
                                  19  11 0001B                 BRB      3$                                                ; 0395
                             0B   A0  95 0001D 1$:             TSTB     11(NAM)                                           ; 0401
                                  0B  13 00020                 BEQL     2$                                                ;
                     6E      0B   A0  9A 00022                 MOVZBL   11(NAM), DESC                                     ; 0404
              04     AE      0C   A0  D0 00026                 MOVL     12(NAM), DESC+4                                   ; 0405
                                  09  11 0002B                 BRB      3$                                                ; 0401
                     6E      34   A1  9A 0002D 2$:             MOVZBL   52(R1), DESC                                      ; 0409
              04     AE      2C   A1  D0 00031                 MOVL     44(R1), DESC+4                                    ; 0410
                     50      0C   AC  D0 00036 3$:             MOVL     FRAB, R0                                          ; 0416
                     7E      08   A0  7D 0003A                 MOVQ     8(R0), -(SP)                                      ;
                             08   AE  9F 0003E                 PUSHAB   DESC                                              ;
                                  01  DD 00041                 PUSHL    #1                                                ;
                             04   AC  DD 00043                 PUSHL    MESSAGE                                           ;
        00000000G    00           05  FB 00046                 CALLS    #5, LIB$SIGNAL                                    ;
                                      04 0004D                 RET                                                        ; 0417

; Routine Size:  78 bytes,    Routine Base:  CODE + 03E9
;  422          0418  1 END
;  423          0419  0 ELUDOM



                                                                                 .EXTRN  LIB$SIGNAL

;                                             PSECT SUMMARY
;
;       Name                        Bytes                              Attributes
;
;  CODE                                  1079  NOVEC,NOWRT,  RD ,  EXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)



;                                       Library Statistics
;
;                                                          -------- Symbols --------        Pages        Processing
;       File                                               Total     Loaded   Percent       Mapped       Time
;
;  _$255$DUA28:[SYSLIB]LIB.L32;1                           18619        126         0       1000           00:01.9




;                                               COMMAND QUALIFIERS

;       BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LIS$:STABACCOP/OBJ=OBJ$:STABACCOP MSRC$:STABACCOP/UPDATE=(ENH$:STABACCOP)

; Size:           1063 code + 16 data bytes
; Run Time:          00:21.6
; Elapsed Time:      01:14.2
; Lines/CPU Min:      1162
; Lexemes/CPU-Min: 35087
; Memory Used:  280 pages
; Compilation Complete
```

0015 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY



STABACCOP
LIS

STAINIT
LIS